**1. PROPUESTA DE SERVICIOS**

**1.1. INTRODUCCIÓN**

El presente documento da respuesta a la convocatoria pública CPTIS-2408-2018 y el pliego de especificaciones PETIS-2408-2018 presentado por la empresa TIS, de esta manera “VENSOFT S.R.L.” presenta una solución a la necesidad de un “SISTEMA DE REGISTROS DE PERFILES DE PROYECTOS FINALES DE GRADO”, definiendo en este documento una visión general del enfoque del desarrollo propuesto.

**1.2 IDENTIFICACIÓN DE LAS NECESIDADES**

Dentro la actividad universitaria y administrativa académica, y como parte de culminante de la formación de profesionales, se tiene la presentación de un proyecto final que permita a los estudiantes obtener su grado académico. Para ello se debe trabajar en un perfil de proyecto, registrarlo, desarrollarlo y una vez aprobado por el tutor y el docente final a través de una carta presentada a los Honorables Consejos de Carrera; se debe proceder a la fase de revisión la misma que está delegada a una comisión de tres personas, las mismas que se denominan tribunales del proyecto, su misión es revisar la idoneidad, cumplimento y satis factibilidad del proyecto.

Cada una de las etapas es importante, ya que al margen de que permite a los estudiantes obtener su grado académico, la información que se administra es de vital importancia sobre todo para la toma de decisiones a nivel administrativo y académico.

La primera etapa se cumple con la elaboración y registro oficial de un perfil de proyecto de grado final, para ello el estudiante debe cumplir con un mínimo de información, como por ejemplo: título, objetivos, etc.

La segunda etapa inicia con la aprobación por parte del tutor(es), que permite la asignación de tribunales para su revisión.

Para el registro del perfil se cuenta con un sistema, que cumple con el objetivo básico de registro, sin embargo, algunos aspectos que han surgido con el tiempo y que inevitablemente han generado cambios, puede que hagan que la información registrada tenga algunos vacíos importantes.

El semestre 1/2018 se ha desarrollado un prototipo de sistema que cubre la segunda etapa, sin embargo, para que el sistema funcione de manera correcta se requiere de información completa y correcta del registro de los perfiles de grado final.

## Alcance del Proyecto

Para dar solución a esta problemática, la empresa **“VENSOFT S.R.L.”** propone el desarrollo de un sistema de información adecuado y que responda a todas las necesidades planteadas. Dicho sistema será desarrollado en base a los requerimientos establecidos en el documento PETIS-2408-2018**.** Las funcionalidades y características del mismo estarán limitadas en función al documento mencionado y a lo establecido en la sección de Servicios Propuestos.

**1.3 SERVICIOS PROPUESTOS**

**“VENSOFT S.R.L**” ofrece sus servicios de desarrollo de software para un

“SISTEMA DE REGISTRO DE PERFILES DE PROYECTOS FINALES DE GRADO”que cumpla con:

* + - Los requerimientos mencionados en **PETIS-2408-2018**
    - Tener una Base de Datos íntegra, segura y fiable.
    - Contar con Interfaces de usuario sencillas y fáciles de usar

**1.4 TECNOLOGÍA A UTILIZAR**

De acuerdo a las características requeridas por el sistema, las tecnologías a usar serán las siguientes, de acuerdo a lo concordado con el equipo de desarrollo después de su análisis:

*Para el código fuente:*

* Se usará como lenguaje de programación base: **PHP**, este es un lenguaje totalmente libre y abierto. Así mismo los desarrolladores de la empresa cuentan con la experiencia necesaria para poder desenvolverse en este lenguaje de una manera eficiente.
* Para la interfaz de usuario se usará **HTML, CSS y Javascript**

*Para la Base de Datos:*

* Se usará el gestor de Base de Datos **MySQL**, por su rapidez y facilidad en el manejo de datos

*En los servidores:*

* Se usará **Apache** en la parte de pruebas como servidor local, por su facilidad de configuración.
* Al momento de implementar el proyecto se usara los servidores del departamento informática y sistemas

*Como parte del control y administración en el proceso de desarrollo se utilizarán las siguientes herramientas:*

* **Git:** Como herramienta de control de versiones.
* **Trello:** Para el seguimiento y control de iteraciones y actividades.

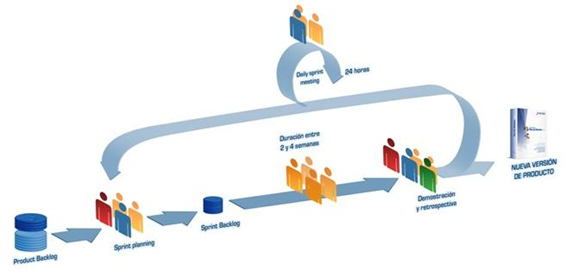
**2. PLANIFICACIÓN DEL PROYECTO**

**2.1. INTRODUCCIÓN**

Este documento describe el marco de trabajo que servirá como guía en el desarrollo de las actividades del proyecto; lo cual permitirá realizar estimaciones razonables en términos de tiempo y recursos que sean necesarios.

**2.2. PROCESO DE DESARROLLO**

El proceso de desarrollo a utilizar tendrá fundamentos en el proceso de desarrollo Scrum. Del cual se tomará y seguirán sus lineamientos y filosofías de desarrollo, adecuándose al contexto propio de la empresa haciendo uso de las ventajas y virtudes de las metodologías ágiles. A dicho proceso de desarrollo se le denominara **“SCRUM”.**



**2.2.1 DESCRIPCIÓN DEL PROCESO DE DESARROLLO**

### **Concepto**

Scrum es una metodología ágil y flexible para gestionar el desarrollo de software, cuyo principal objetivo es maximizar el retorno de la inversión para su empresa. Se basa en construir primero la funcionalidad de mayor valor para el cliente y en los principios de inspección continua, adaptación, auto-gestión e innovación.

### **Beneficios**

* **Cumplimento de expectativas:** El cliente establece sus expectativas indicando el valor que le aporta cada requisito / historia del proyecto, el equipo los estima y con esta información el Product Owner establece su prioridad. De manera regular, en las demos de Sprint el Product Owner comprueba que efectivamente los requisitos se han cumplido y transmite se feedback al equipo.
* **Flexibilidad a cambios:** Alta capacidad de reacción ante los cambios de requerimientos generados por necesidades del cliente o evoluciones del mercado. La metodología está diseñada para adaptarse a los cambios de requerimientos que conllevan los proyectos complejos.
* **Reducción del Time to Market:** El cliente puede empezar a utilizar las funcionalidades más importantes del proyecto antes de que esté finalizado por completo.
* **Mayor calidad del software:** La metódica de trabajo y la necesidad de obtener una versión funcional después de cada iteración, ayuda a la obtención de un software de calidad superior.
* **Mayor productividad:** Se consigue entre otras razones, gracias a la eliminación de la burocracia y a la motivación del equipo que proporciona el hecho de que sean autónomos para organizarse.
* **Maximiza el retorno de la inversión (ROI):** Producción de software únicamente con las prestaciones que aportan mayor valor de negocio gracias a la priorización por retorno de inversión.
* **Predicciones de tiempos:** Mediante esta metodología se conoce la velocidad media del equipo por sprint (los llamados puntos historia), con lo que consecuentemente, es posible estimar fácilmente para cuando se dispondrá de una determinada funcionalidad que todavía está en el Backlog.
* **Reducción de riesgos:** El hecho de llevar a cabo las funcionalidades de más valor en primer lugar y de conocer la velocidad con que el equipo avanza en el proyecto, permite despejar riesgos eficazmente de manera anticipada.

### **El proceso**

El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada Sprint, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. En cada nuevo Sprint, se va ajustando la funcionalidad ya construida y se añaden nuevas prestaciones priorizándolos siempre aquellas que aporten mayor valor de negocio

* **Product Backlog:** Conjunto de requisitos denominados historias descritos en un lenguaje no técnico y priorizados por valor de negocio, o lo que es lo mismo, por retorno de inversión considerando su beneficio y coste. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares.
* **Sprint Planning:** Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese sprint, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir.
* **Sprint:** Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo.
* **Sprint Backlog:** Lista de las tareas necesarias para llevar a cabo las historias del sprint.
* **Daily sprint meeting:** Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada. Cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos.
* **Demo y retrospectiva:** Reunión que se celebra al final del sprint y en la que el equipo presenta las historias conseguidas mediante una demostración del producto. Posteriormente, en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos.

### **Roles**

La gestión de un proyecto Scrum se centra en definir cuáles son las características que debe tener el producto a construir y está formado por los siguientes roles:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| * **Scrum master:** Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el Product Owner para maximizar el ROI. * **Product owner (PO):** Representante de los accionistas y clientes que usan el software. Se focaliza en la parte de negocio y él es responsable del RoI del proyecto (entregar un valor superior al dinero invertido). Traslada la visión del proyecto al equipo, formaliza las prestaciones en historias a incorporar en el Product Backlog y las re prioriza de forma regular. * **Team: Grupo** de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.   **2.3 PLANIFICACIÓN**   |  |  | | --- | --- | | Etapa del proceso de desarrollo | Actividad planteada | | Sprint 0 | * Análisis, diseño preliminar del problema planteado * Preparación y configuración del entorno de desarrollo: Hardware y software * Establecimiento de las herramientas de gestión a utilizar * Capacitación del equipo | |  | 1. Registro información básica del perfil 2. Registro áreas perfil 3. Registro de fechas del perfil 4. Registrar modalidad perfil 5. Número de personas del perfil 6. Registrar tutor del proyecto 7. Registrar docente de la materia 8. Registrar estado de un perfil 9. Registrar carrera(s) del perfil 10. Formulario de acuerdo a la modalidad 11. Permitir la Inserción de áreas de conocimiento 12. Permitir la Inserción de un profesional para ser tutor 13. Proveer mecanismo que permitan aceptar la renuncia de un tutor 14. Registrar datos relativos a la actualización del perfil 15. Reportar información requerida para fines de administración 16. Registrar datos relativos al cambio del perfil 17. Normalizar los datos ya existentes 18. Proveer de un mecanismo de alerta respecto a tiempo del perfil 19. Migrar datos ya existentes 20. registro de asesor si corresponde 21. Privilegiar tareas del sistema de acuerdo a roles 22. Proveer de una descripción básica de los pasos a seguir previos al registro de proyectos, 23. Crear cuentas a estudiantes 24. Permitir eliminación de formularios, bajo privilegios 25. integrar y/o proveer de un mecanismo de conexión con el prototipo desarrollado en la gestión 1/2018 26. manejo de banco de datos por gestiones |   **2.4 CRONOGRAMA DE ACTIVIDADES**  A continuación, en la Tabla se observa el cronograma de actividades.   |  |  |  |  | | --- | --- | --- | --- | | # Sprint | Fecha comienzo | Fecha finalización | Duración | | Sprint 0 | 1/09/2018 | 10/09/2018 | 10 días | | Sprint 1 | 11/09/2018 |  |  | | Sprint 2 |  |  |  | | Sprint 3 |  |  |  | | Sprint 4 |  |  |  | |  |  |  |  | |

**3. PLAZO DE CONCLUSIÓN DEL CONTRATO**

Considerando entregas parciales de acuerdo al cronograma de actividades, se ha establecido como fecha de entrega del producto final según **CPTIS-2408-2018** hasta el día viernes 7 de diciembre del 2018

**4. PROPUESTA ECONÓMICA**

**4.1. ESTIMACIÓN DE COSTOS DE PROYECTO**

Se realizó la estimación de costos del proyecto en base a los **PUNTOS DE FUNCIÓN** dependiendo de la complejidad de los mismos

|  |  |
| --- | --- |
| Tipo de Función | Descripción |
| Entrada Externa (EI) | Pantallas donde los usuarios ingresan datos |
| Salida Externa (EO) | Informes, gráficos, listado de datos |
| Consulta Externa (EQ) | Recuperar y mostrar datos del usuario (Buscar) |
| Archivo Lógico Interno (IFL) | Archivo del punto de vista lógico, no como en un sistema operativo, puede ser tablas en la base de datos |
| Archivo de Interfaz Externo (EIF) | Datos referenciados a otros sistemas, datos mantenidos por otros sistemas peros usado por el sistemas actual |

Cálculo de PFNA (Puntos de función no ajustados).

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tipo función | complejidad | | | Complejidad esperada | peso | Complejidad esperada |
| **simple** | **medio** | **complejo** |
| (EI) Entrada Externa | 3 | 4 | 6 | 6 | 13 | 78 |
| (EO) Salida Externa | 4 | 5 | 7 | 5 | 5 | 25 |
| (EQ) Consulta Externa | 3 | 4 | 6 | 6 | 6 | 36 |
| (ILF) Archivo Lógico Interno | 7 | 10 | 15 | 10 | 5 | 50 |
| (EIF) Archivo de Interfaz externo | 5 | 7 | 10 | 10 | 3 | 30 |
| TOTAL PF NO AJUSTADOS (PFNA) | | | | | | 219 |

**4.2. CALCULO DE LA COMPLEJIDAD**

Para el cálculo de ajuste de complejidad se tomará en cuenta la siguiente puntuación:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ausente | Insignificante | Moderada | Media | Significante | Fuerte |
| 0 | 1 | 2 | 3 | 4 | 5 |

Grado de influencia del factor de la complejidad:

|  |  |  |
| --- | --- | --- |
|  | factor | Puntaje |
| 1 | Comunicación de datos | 5 |
| 2 | Procesamiento distribuido | 3 |
| 3 | Objetivos de Rendimiento | 3 |
| 4 | Configuración del equipamiento | 2 |
| 5 | Tasa de transacciones | 2 |
| 6 | Entrada de datos en línea | 5 |
| 7 | Interface con el usuario | 5 |
| 8 | Actualización en línea | 3 |
| 9 | Procesamiento complejo | 3 |
| 10 | Reusabilidad del código | 4 |
| 11 | Facilidad de implementación | 3 |
| 12 | Facilidad de operación | 4 |
| 13 | Instalaciones Múltiples | 4 |
| 14 | Facilidad de cambios | 5 |
| ΣFi | | 51 |

PF=PFNA\*(0.65+0.01\*ΣFi)

PF=219\*(0.65+0.01\*51)=254

Según los cálculos realizados se tendrán 266 PUNTOS DE FUNCIÓN en el software a desarrollar.

**4.3. PRODUCTIVIDAD**

El tiempo estimado para el desarrollo del software será de 3 meses, tomando en cuenta de lunes a viernes, durante 13 semanas a partir de la firma del contrato, se considerará como una jornada de trabajo 3 horas diarias y el número de miembros a trabajar es de 4 personas en el equipo de trabajo.

Para realizar el cálculo de productividad por meses se tomará en cuenta 3 meses de trabajo a partir de la fecha de firma del contrato

**PRODUCTIVIDAD = (254 [PF]/4 personas) / 3meses = 21[PF/ Persona]**

**4.4. COSTOS DE RECURSOS HUMANOS**

|  |  |  |  |
| --- | --- | --- | --- |
| Costo Hora de Trabajo | Horas Trabajadas al Día | Días Trabajados a la Semana | Total en Semanas de Trabajo |
| 10 $us | 3 horas | 6 días | 13 semanas |

**Sueldo del Programador = 10 \* 3 \*6 \* 13 = 2340 $us.**

**Sueldo del Programador por Mes = 2340 / 3 = 780 $us.**

Tomando en cuenta los puntos de Función, el sueldo del programador y el esfuerzo requerido en el desarrollo del software se llegó a los siguientes resultados:

**Costo de Cada PF = Costo Persona Mes / Productividad**

**Costo de Cada PF = 780/21= 37.14**

**Costo Total = PF \* Costo de Cada PF**

**Costo Total = 266 \* 37.14 = 9879.24$us** ≈ **9880 $us.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PF | Productividad Esperada (PF/Persona) | Costo Persona / Mes | Costo de cada PF | Costo Total |
| 266 | 21 | 780 | 37.14 | 9880 |

**4.5. COSTOS OPERATIVOS**

En la siguiente tabla se detalla el costo operativo mensual:

|  |  |
| --- | --- |
| Descripción | Monto |
| Luz | 20 $ |
| Agua | 10 $ |
| Internet | 30 $ |
| Alquileres | 250 $ |
| Mantenimiento | 20 $ |
| Otros | 20 $ |
| TOTAL | **350 $** |

Desglosando el Costo operativo mensual a costo operativo semanal (para esto suponemos que un mes comercial tiene 4 semanas), se tendría:

350 / 4 = 87,5 $us/semana

Después de determinar nuestro costo operativo semanal obtendremos el costo operativo en el transcurso del desarrollo del software:

87,5 \* 13 =1137.5 ≈ 1138 $us.

**4.6. COSTOS DE CAPACITACIÓN A USUARIOS FINALES**

Los costos de capacitación serán expuestos en la siguiente tabla:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ítem | Días | Horas por día | Personas | Costo por hora | Costo total |
| Desarrollo de Manual de Usuario para El manejo del Software | 2 | 3 | 1 | 10 | 60 |
| Total | | | | | 60 |

**4.7. COSTO TOTAL**

Teniendo en conocimiento los costos de recursos humanos, costos operativos tendremos el siguiente costo total sin incluir las utilidades de la empresa:

**Costo Total = Costo RRHH + Costo Operativo**

**Costo Total = 9880+ 1138**

**Costo total = 11018 $us.**

En total la estimación de costos expresada en $us y bolivianos será la siguiente:

|  |  |  |
| --- | --- | --- |
|  | Costo en $ | Cocto en Bs t/c 6.96 |
| Costo Recursos Humanos | 9880 | 98764.4 |
| Costo Operativo | 1138 | 7920.48 |
| Costos de capacitación a usuarios | 60 | 417.6 |
| Costo Total | 11078 | 77102.88 |

El costo total de proyectos según la estimación realizada asciende a 11078 [$] (Once mil setenta y ocho $) este monto expresado en bolivianos ascendería a 77102.88 Bs (Setenta y siete mil ciento dos 88/100 Bs)

**4.8. PLAN DE PAGOS**

El plan de pagos se puede negociar con el cliente. La Empresa VENSOFT S.R.L. a manera de facilitar el trabajo con la Empresa "TIS" plantea la siguiente propuesta: Realizar pagos parciales a medida que se vayan concluyendo los Sprints, definiendo las cuotas para cada pago.

La forma de pago tentativa y los montos estimados, se muestran en la Tabla.

|  |  |  |
| --- | --- | --- |
| Descripción | Porcentaje (%) | Monto (Bs) |
| 1 Sprint de desarrollo | 20 |  |
| 2 Sprint de desarrollo | 25 |  |
| 3 Sprint de desarrollo | 25 |  |
| 4 Sprint de desarrollo | 30 |  |
|  |  |  |
| TOTAL | 100 | 77102.88 |

**4.9. ANÁLISIS DE RIESGOS**

La empresa VENSOFT S.R.L. clasifica el impacto de la siguiente forma:

|  |
| --- |
| Impacto |
| Baja |
| Media |
| Moderada |
| Alta |

**4.9.1. RIESGOS TECNOLÓGICOS**

En esta categoría mencionaremos los riesgos de alguna falla de las herramientas de trabajo, tanto como el hardware, base de datos, los componentes que se reutilizarán, etc.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Código | Riesgo | Probabilidad Ocurrencia | Impacto | Causas probables |
| R1 | - Las Computadoras de trabajo presentan anomalías en su funcionamiento | 30% | MEDIO | -Infección de Virus  -Falta de Mantenimiento |
| R2 | -Posibles pérdidas de la información | 30% | MODERADO | -Cortes de Luz  -Falta de Backups |
| R3 | -Vulnerabilidad de la seguridad del software | 20% | BAJO | -Ingreso de Intruso al Software |
| R4 | -Plataforma y software del equipo de desarrollo distintos a los del servidor | 40% | ALTO | -No se toma en cuenta la configuración de la plataforma tanto en software como en hardware |

**4.9.2. RIESGOS ASOCIADOS AL EQUIPO DE TRABAJO**

En cuanto a esta categoría de los riesgos asociados al equipo de trabajo mencionaremos los riesgos que el personal ocasiona al grupo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Código | Riesgo | Probabilidad Ocurrencia | Impacto | Causas probables |
| R5 | - Falta de conocimiento o capacitación para desarrollar el proyecto | 35% | ALTO | -Poco interés de los integrantes en capacitarse  -Personal inexperto en el desarrollo de software |
| R6 | -Baja moral del personal, falta de confianza individual o grupal | 30% | MODERADO | -Nunca se desarrolló un proyecto similar  -Temor a los retos desconocidos |
| R7 | -Abandono del personal comprometido durante el desarrollo del software | 50% | ALTO | -Falta de seriedad y responsabilidad de los integrantes  -Causa externa más allá del alcance de los integrantes |

**4.9.3. RIESGOS RELACIONADOS CON LOS REQUERIMIENTOS DEL CLIENTE**

Se podría presentar en este punto una mala interpretación de los principales requerimientos del software debido a cambios con alto impacto en el diseño

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Código | Riesgo | Probabilidad Ocurrencia | Impacto | Causas probables |
| R8 | - No tiene una idea clara de lo que quiere | 45% | ALTO | -Cliente Indeciso  -Falta de Comunicación con el cliente |
| R9 | -Requerimientos demasiados complejos | 35% | ALTO | -No se hace saber al cliente sobre las limitaciones de software |
| R10 | -Cambios o adición de nuevos requerimientos | 30% | MODERADO | -No se logró capturar de manera óptima los requisitos del cliente |

**4.9.4. RIESGOS RELACIONADOS CON LA ESTIMACIÓN**

Uno de los principales problemas en este punto seria tomar en cuenta la falta de financiamiento del proyecto puesto que este factor es de suma importancia para el desarrollo de este. Además, se tomará muy en cuenta también el factor del tiempo y dinero estimado.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Código | Riesgo | Probabilidad Ocurrencia | Impacto | Causas probables |
| R11 | - Subestimación del tiempo de duración total del proyecto | 35% | ALTO | -Mala Planificación de actividades |
| R12 | -Subestimación a cerca de los recursos a utilizarse en dinero | 20% | MEDIO | -Mala Planificación de estimación de costos del proyecto |

**4.9.5. PLAN DE CONTINGENCIA CONTRA RIESGOS**

Para evitar los riesgos presentados anteriormente se tiene un plan para evitar que afecten al Producto de software

|  |  |  |
| --- | --- | --- |
| Código | Riesgo | Plan de contingencia |
| R1 | * Las Computadoras de trabajo presentan anomalías en su funcionamiento | * Instalar un antivirus * Realizar mantenimiento * No usar herramientas de fuentes desconocidas |
| R2 | * Posibles pérdidas de la información | * Realizar backups * Utilizar un repositorio en la nube (Github) |
| R3 | * Vulnerabilidad de la seguridad del software | * Que solo los individuos involucrados tengan acceso al software |
| R4 | * Plataforma y software del equipo de desarrollo distintos a los del servidor | * Preguntar al cliente los requerimientos del servidor |
| R5 | * Falta de conocimiento o capacitación para desarrollar el proyecto | * Usar herramientas, lenguajes, framaworks conocidos o fáciles de aprender |
| R6 | * Baja moral del personal, falta de confianza individual o grupal | * Tener metas claras * Mejor comunicacion |
| R7 | * Abandono del personal comprometido durante el desarrollo del software | * Tener compromiso con la empresa |
| R8 | * No tiene una idea clara de lo que quiere | * Conversar con el cliente * El cliente debe ser parte del desarrollo |
| R9 | * Requerimientos demasiados complejos | * Dividir los requerimientos en unos más específicos |
| R10 | * Cambios o adición de nuevos requerimientos | * Conversar con el cliente * El cliente debe ser parte del desarrollo |
| R11 | * Subestimación del tiempo de duración total del proyecto | * Planificar de manera correcta |
| R12 | * Subestimación a cerca de los recursos a utilizarse en dinero | * Conocer las capacidades de los miembros del equipo |